REMARKS

A substitute specification is submitted herewith that is believed to address the specification objections noted in the Official Action.

Replacement drawing figures are submitted for Figures 2, 14A, 14B, 15A and 15B and are believed to address the drawing objections noted in the Official Action. Neither the substitute specification nor the replacement drawing figures present new matter.

Claims 1-21 were previously pending in the application. Claims 1-21 are canceled and replaced with new claims 22-35. The new claims are believed to address the claim objections noted in the Official Action.

Claims 1-21 are rejected as unpatentable over applicant's disclosed prior art in view of TRAN 5,765,035.

Reconsideration and withdrawal of the rejection are respectfully requested because the references do not teach or suggest a first comparator circuit comparing, in view of an index and an offset, an object instruction with tag-retrieval to at least one store instruction stored in a store queue, and a second comparator circuit comparing, in view of an index and a way, an object instruction with tag-retrieval to at least one store instruction stored in the store queue, as recited in new claim 22 of the present application.

By way of example, the first and second comparator circuits compare, in view of index, off-set and way, the object instruction with tag-retrieval to the at least one store instruction stored in the store queue for avoiding any unnecessary stalls. For example, a first unnecessary stall to the object instruction with tag-retrieval and a second unnecessary stall to the object instruction with tag-retrieval if the object instruction with tag-retrieval does not correspond to or is not of any of the store instructions in the store queue. The first unnecessary stall is generated based on a dependency between the object instruction with tag-retrieval and the store instruction. The second unnecessary stall is generated upon replacing a cache.

The first comparator circuit compares, in view of index and off-set, the object instruction with tag-retrieval to all store instructions stored in the store queue. The first comparator circuit outputs a first stall signal for stalling the object instruction with tag-retrieval if the object instruction with tag-retrieval corresponds, in view of index and off-set, to at least any one of the store instructions stored in the store queue. The correspondence, in view of index and off-set, between the object instruction with tag-retrieval and a store instruction represents or means that there is any dependency relationship between the object instruction with tag-retrieval and the store

instruction, whereby the object instruction with tag-retrieval is placed into a first stall state. The first comparison by the first comparator circuit is made with reference not only to the index but also to the off-set. If the object instruction with tag-retrieval corresponds, in view of index only, to the store instruction but does not correspond, in view of off-set, to the store instruction, then the object instruction with tag-retrieval is not placed into the first stall state. This means avoiding the unnecessary first stall to the object instruction with tag-retrieval.

The second comparator circuit compares, in view of index and way, the object instruction with tag-retrieval to all of the store instructions stored in the store queue for the purpose of verifying or confirming whether a cache line related to the object instruction has already been present or yet been absent in the cache as the cache line of the store instruction, for which reason the object instruction is not limited to the store instruction.

If the cache line is present in the cache, then the cache becomes replaceable but after the store instruction in the store queue has been executed. Namely, the object instruction is placed in the second stall state, and the store instruction in the store queue is executed before the cache is the replaced.

If the cache line is absent in the cache, then only replacement of the cache is made. In this case, not only the index but also the way are subject to the comparison, thereby avoiding the second unnecessary stall which is caused when a mistake is made of the presence of the cache line in the cache, even if the cache line is actually absent in the cache.

In accordance with the applicant's disclosed prior art shown in the drawing of FIG. 5, only the index is subject to the comparison (S101). It is, in this case, possible that the first stall is unnecessary. Taking into account that not only index but also off-set are subject to the comparison, the first unnecessary stall is avoidable, while the second unnecessary stall is unavoidable.

As one example, it is assumed that two store instructions being identical with each other in view of index and being different from each other in view of off-set are present in the store queue, and the cache is two-ways. If an object instruction is identical in view of index only with the two store instructions and is different in view of off-set from both the two store instructions, then no correspondence in off-set is confirmed, whereby without execution of the store instruction, the cache is replaced and the object instruction is executed, whereby the object instruction is not placed in the second stall state, even if it should have been necessary that the object

instruction be placed in the second stall state. Accordingly, applicant's disclosed prior art is not able to avoid unnecessary stalls.

TRAN teaches that the address-comparison is made in order to detect a dependency relationship from store instruction to load instruction. Changing the execution sequence of instructions depends upon the address dependency relationship, for which reason the store instruction is compared, in view of index and off-set, to the load instruction. TRAN is irrelevant to a process for confirming whether the cache line is present or absent in the cache and also irrelevant to a process for replacing the cache. TRAN is unable to accurately and correctly execute the second stall to place the object instruction into the second stall state.

In accordance with new independent claims 22 and 30, not only index but also way are subject to a comparison in order to detect a dependency relationship from store instruction in the store queue to the presence, in the cache, of the cache line subject to the store. It is not an object of the present invention to detect the dependency relationship between the store instruction and the load instruction like in TRAN. The present invention makes the address-comparison for monitoring whether the cache line subject to the store is replaced from the cache by any undesired influence of other instruction.

In addition, TRAN takes a data flow direction from Data Cache to Stack Cache. Stack Cache is used on the Decode state.

In contrast, the present invention takes an opposite data flow direction from Store Queue to Data Cache. Store Queue is used on WriteBack stage. Therefore, one of ordinary skill in the art would not look to TRAN to provide teachings to render obvious the present claims.

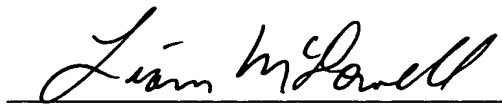
Accordingly, the new claims are believed to avoid the rejection under §103 and are allowable over the art of record.

In view of the present amendment and the foregoing remarks, it is believed that the present application has been placed in condition for allowance. Reconsideration and allowance are respectfully requested.

The Commissioner is hereby authorized in this, concurrent, and future replies, to charge payment or credit any overpayment to Deposit Account No. 25-0120 for any additional fees required under 37 C.F.R. § 1.16 or under 37 C.F.R. § 1.17.

Respectfully submitted,

YOUNG & THOMPSON



Liam McDowell, Reg. No. 44,231
745 South 23rd Street
Arlington, VA 22202
Telephone (703) 521-2297
Telefax (703) 685-0573

LM/mjr
June 8, 2004